Memory Management Proficiency

Week-01:

Day	Topics	Tasks
01	Stack vs Heap vs Global Memory	- Study memory types: Stack, Heap, Static (Global). - Write small programs to print addresses of different variables.
02	Pointers & References	- Learn how to declare and use pointers and references. - Modify data through pointers. - Pointer to Pointer concepts.
03	Dynamic Memory Allocation (new / delete)	- Practice allocating and deallocating memory using new and delete. - Study what happens when you forget to deallocate (memory leaks).
04	Stack Overflow & Memory Leak Simulation	- Write a recursive function that causes a Stack Overflow. - Write a program that leaks memory intentionally.
05	Fixing Memory Leaks Properly	- Refactor the previous day's code to correctly deallocate memory. - Practice both delete and delete[].
06	Stack vs Heap Arrays	- Create arrays on Stack and Heap. - Compare behaviors (lifetime, access speed). - Measure time if possible.
07	Weekly Review	- Rewrite main programs without looking. - Identify why each error happens and how you solved it.

Week-02:

Day	Topics	Tasks
08	Stack Frames Visualization	- Trace how local variables are arranged in memory. - Manually simulate Stack Frames.
09	Heap Fragmentation Simulation	- Write a program to simulate Heap Fragmentation (allocate/free blocks randomly).
10	Best Practices in Manual Memory Management	- Study rules like: "Who allocates, deallocates," avoiding double-free, etc. - Create a small checklist for memory-safe code.
11	Memory Debugging Tools (Valgrind /Address Sanitizer)	- Install a memory checker. - Learn how to detect leaks and invalid memory access.
12	Practical Debugging Session	- Create a small buggy program. - Find and fix the memory issues using debugging tools.
13	Dynamic Linked List Project	- Build a full Linked List with manual new/delete. - Validate memory usage with tools.
14	Final Project: Mini Memory Pool	- Create a simple memory pool or allocator. - Focus on minimizing fragmentation and manual memory handling.

Daily Time Distribution:

Time	Activity	Description
1.5 hours	Study and Understanding	Watch or read high-quality resources. Take notes manually.
1.5 hours	Coding Practice	Implement the concept you studied immediately. No copy-paste.
0.5 hours	Review / Experiment	Repeat the code without looking OR explore extra scenarios.